

UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Tambahan

Sidang 1988/89

Jun 1989

CSS101 - Penghantar Sistem Komputer

Masa : [3 jam]

---

Sila pastikan bahawa kertas peperiksaan ini mengandungi 8 muka surat yang bercetak sebelum anda memulakan peperiksaan ini.

Jawab SEMUA soalan.

Semua nombor yang digunakan di dalam konteks mesin PDP-II atau aturcara MACRO-II adalah di dalam peralapanan melainkan jika dinyatakan sebaliknya.

Semua jawapan mestilah ditulis di dalam Bahasa Malaysia.

1. (a) Nyatakan tiga sebab kenapa bahasa penghimpun (assembly) lebih mudah digunakan berbanding dengan bahasa mesin.

[10/100]

- (b) Kecekapan aturcara bahasa assembly boleh diukur melalui bilangan capaian dan simpanan yang diperlukan untuk melaksanakan sesuatu arahan.

(i) Terangkan dengan ringkas konsep capaian dan simpanan.

(ii) Berapakah bilangan capaian dan simpanan yang diperlukan untuk melaksanakan setiap arahan berikut:

```
MOV    R1 , A
      dan
ADD    #2 , B
```

[15/100]

...2/-

(c) Bagaimanakah nombor negatif diwakilkan oleh PDP-II.

Namakan dua lagi perwakilan bagi nombor negatif yang boleh diamalkan oleh sesuatu sistem komputer.

[15/100]

(d) Aturcara berikut bertujuan untuk meletakkan 1 ke dalam P jika (X) kurang atau sama dengan sifar dan meletakkan 1 ke dalam Q jika (X) lebih besar atau sama dengan sifar; tetapi ia tidak berbuat demikian. Terangkan mengapa dan berikan aturcara yang betul.

Nota: Maksud (X) ialah kandungan X.

```

                TST X
                BGT L1
                MOV #1 , P
L1 :           BLT L2
                MOV #1 , Q
L2 :           HALT
```

[20/100]

(e) Surih aturcara berikut:

```

L1 :           DEC A
                BEQ L1
                BR L2
                .BLKW 5

L2 :           MOV #A , R1
                MOV A , R2
                HALT

A :           .WORD 10
```

(i) Sekiranya aturcara ini bermula di 1000<sub>8</sub> ; apakah nilai yang diberi kepada A? Berapa perkataan yang diperlukan untuk menyimpan kod mesin untuk aturcara ini?

(ii) Apakah kandungan R1 dan R2 selepas pelaksanaan HALT?

[20/100]

...3/-

- (f) (i) Lukiskan gambarajah untuk senibina satu sistem komputer PDP-II. Namakan setiap bahagian.
- (ii) Lukiskan komponen untuk unit pemprosesan pusat PDP-II. Namakan setiap bahagian.

[20/100]

2. (a) Nyatakan perbezaan-perbezaan yang terdapat di antara pernyataan-pernyataan berikut:

(i)	SUB A , B	dan	CMP A , B
(ii)	.BLKW n	dan	.BLKB n
(iii)	.WORD n	dan	.BYTE n
(iv)	Daftar	dan	Lokasi memori
(v)	Arahan	dan	Perintah Penghimpun

[15/100]

- (b) Diberi maklumat berikut:

(A)	=	070707
(B)	=	107070
(C)	=	123456
(D)	=	054321

Apakah kesan kepada kandungan A, B, C dan D serta bit kod kondisi N, Z, V dan C selepas pelaksanaan setiap arahan berikut. Gunakan kandungan A, B, C dan D yang asal bagi setiap arahan.

(i)	BIS B,C	(ii)	BIT C,C
(iii)	BIC A,C	(iv)	BIS A,B
(v)	BIC C,A		

[15/100]

- (c) Nyatakan rumusan yang boleh dibuat mengenai bit kod kondisi C dan V dalam penambahan nombor-nombor pelengkap-dua.

- (i) Jika kedua-dua nombor adalah positif
- (ii) Jika kedua-dua nombor adalah negatif
- (iii) Jika satu nombor positif dan satu nombor negatif.

[15/100]

...4/-

- (d) Keputusan untuk pelaksanaan arahan CMP X,Y tidak disimpan, arahan perbandingan ini selalu diikuti oleh arahan cabang. Bagaimanakah sesuatu arahan cabang dapat dilaksanakan sekiranya keputusan dari CMP X,Y ini tidak disimpan?

[15/100]

- (e) Tulis satu aturcara yang lengkap dalam MACRO-II untuk mengira hasilbahagi dua nombor melalui penolakan. Pastikan kedua-dua nombor ini adalah nombor positif.

[25/100]

- (f) Apakah nilai kandungan R1 selepas pelaksanaan arahan-arahan berikut:

```
MOV    #-1,  R1
ASR    R1
ASR    R1
ASR    R1
ROL    R1
```

[15/100]

3. (a) Berikut ialah bahagian data untuk satu aturcara.

```
      . PSECT DATA
ARRAY : . BLKW  1000
TAMAT : . WORD
```

ARRAY ialah satu susunertib satu dimensi. Berikan nilai awal sifar kepada setiap unsur ARRAY. Tuliskan kod yang berasingan untuk setiap mod pengalamatan berikut:

- (i) tertangguh
- (ii) autobertambah
- (iii) autoberkurang
- (iv) indeks

[20/100]

...5/-

- (b) Anda dikehendaki untuk menjumlahkan dua nombor yang disimpan di lokasi memori A dan B. Hasil jumlah disimpan di dalam JUMLAH. Jika berlaku ralat limpah untuk penjumlahan ini, lokasi RLIM mengandungi 1. Jika tiada ralat limpah, RLIM mengandungi 0.

Tulis subrutin untuk tujuan di atas dengan A dan B sebagai hujah masukan serta RLIM dan JUMLAH sebagai hujah keluaran. Gunakan teknik penyiaran hujah berikut:

- (i) Teknik lokasi tetap
- (ii) Teknik panggilan stek dengan menyimpan alamat hujah di dalam stek
- (iii) Teknik alamat berjujukan

[30/100]

- (c) Untuk setiap teknik penyiaran hujah yang anda gunakan dibahagian (b), huraikan kelebihan serta kekurangan masing-masing.

[15/100]

- (d) Diberi takrifan makro berikut:

```
. MACRO    BAHAGI    Y , K
          ASR Y
KOUNT = KOUNT + 1
. IF NE,KOUNT - K
  BAHAGI    Y,K
.ENDC
.ENDM
```

Tunjukkan pengembangan makro selepas panggilan berikut dibuat:

```
KOUNT = 0
BAHAGI    R0 , 10
```

Terangkan dengan ringkas dan tepat kesan panggilan di atas.

[15/100]

...6/-

- (e) Surih pelaksanaan aturcara berikut dengan menunjukkan kandungan R3, R4, SP dan stek selepas pelaksanaan setiap arahan.

```
LC = .  
. = 500 + LC  
MULA : MOV      PC,SP  
        TST     - (SP)  
        MOV     #123,R3  
        JSR    R3, SUB  
        .WORD  5  
        HALT  
SUB :   MOV     (R3)+,R4  
        DEC    R4  
        BEQ    TAMAT  
        JSR    R3,SUB  
        .WORD  1  
TAMAT: RTS R3  
        .END MULA
```

[20/100]

4. (a) Terangkan perbezaan di antara arahan-arahan berikut:

```
(i)   JSR PC,SUB & JSR R5,SUB  
(ii) RTS PC    & RTS R5
```

[20/100]

- (b) Berikut ialah tempahan storan untuk satu stek bait di dalam bahasa MACRO-II

```
.BLKB 7  
STEK: .BLKB 1
```

Tuliskan arahan MACRO-II untuk melaksanakan perkara berikut:

- (i) Jadikan R5 sebagai petunjuk stek.  
(ii) Pindahkan kandungan bait rendah R1 diikuti oleh kandungan bait tinggi R1 ke dalam stek. (Gunakan R5 sebagai petunjuk stek).

...7/-

- (iii) Keluarkan unsur teratas stek dan simpan di dalam bait beralamat A. (Gunakan R5 sebagai petunjuk stek).

[20/100]

- (c) Diberi kandungan daftar serta lokasi memori berikut:

Daftar	Kandungan	Alamat	Kandungan
R0	000002	000500	000502
R1	000500	000502	000504
R2	000504	000504	000506
R3	000506	000506	000510

Nyatakan samada arahan-arahan berikut sah atau tidak. Jika sah, tunjukkan kandungan daftar/lokasi memori yang berubah selepas perlaksanaan arahan tersebut. Jika tidak sah, beri penjelasan kenapa ianya tidak sah.

- (i) `MOVB R1, R0`
- (ii) `CLR @-(R3)`
- (iii) `CLRB -(R2)`
- (iv) `ADD @ (R1)+, R1`
- (v) `DIV #20, (R2)+`

[25/100]

- (d) Tukarkan setiap arahan di (c) ke kod mesinnya yang setara (jika ada).

[5/100]

- (i) Apakah perbezaan utama di antara arahan perkataan dan arahan bait
- (ii) Adakah setiap arahan perkataan mempunyai arahan baitnya yang setara? Jelaskan jawapan anda.

[15/100]

...8/-

- (e) Huraikan dengan ringkas dan tepat kitar pelaksanaan arahan untuk aturcara di bawah. Nyatakan juga bilangan capaian yang diperlukan untuk perlaksanaannya.

```
MOV # 2 , R1
ADD # 3 , R1
HALT
```

[15/100]

-ooo00ooo-



ARAHAN-ARAHAN PDP-11

D = Kod 6-bit untuk operan matlamat      SS = Kod 6-bit untuk operan sumber  
 R = Kod 3-bit untuk daftar  
 X = 0 untuk arahan perkataan, 1 untuk arahan bait

**ARAHAN 1- OPERASI**

	<u>Nemonik</u>	<u>Kod mesin</u>	<u>Kesan</u>	<u>Bit kod kondisi yang berubah</u>
	CLR(B)	x050DD	clear dst	N,V,C clr; Z set
	DEC(B)	x053DD	subtract 1 from dst	N,Z,V set/clr by result
	INC(B)	x052DD	add 1 to dst	N,Z,V set/clr by result
	NEG(B)	x054DD	negate dst	N,Z,V set/clr by result
<b>OPERAND</b>	TST(B)	x057DD	set condition codes	C clr if result = 0, else set N,Z set/clr by dst contents V,C cleared
	COM(B)	x051DD	complement dst	N,Z set/clr by result, V clr, C set
	ASR(B)	x062DD	shift dst 1 place right replicate high-order bit	N,Z set/clr by result C ← old low-order bit of dst V ← exclusive OR of N, C bits
	ASL(B)	x063DD	shift dst 1 place left put 0 in low-order bit	N,Z, set/clr by result C ← old high-order bit of dst V ← exclusive OR of N,C bits
<b>SINGLE</b>	ADC(B)	x055DD	add C bit to dst	all set/clr by result
	SBC(B)	x056DD	subtract C bit from dst	all set/clr by result
	SXT	0067DD	all dst bits to value of N bit	Z set if N bit clr, V clr
	ROR(B)	x060DD	rotate dst right 1 bit, via C bit	N,Z set/clr by result C ← old low-order bit of dst V ← exclusive OR of N,C bits
	ROL(B)	x061DD	rotate dst left 1 bit, via C bit	N,Z set/clr by result C ← old high-order bit of dst V ← exclusive OR of N,C bits
	SWAB	0003DD	swap dst bytes	N,Z set/clr by old dst value V,C clr

**ARAHAN CABANG**

	<u>Nemonik</u>	<u>Kod</u>	<u>Syarat</u>	<u>Nemonik</u>	<u>Kod</u>	<u>Syarat</u>
<b>BRANCHES</b>	BR	000400	always	BHE	001000	Z = 0
	BEO	001400	Z = 0	BEL	100000	N = 0
	BMI	100400	N = 1	BEC	103000	C = 0
	BOS	103400	C = 0	BVO	102000	V = 0
	BVS	102400	V = 1	BGE	002000	$N \vee V = 0$
	BLT	000400	$N \vee V = 1$	BGT	003000	$Z \vee (N \vee V) = 0$
	BLE	003400	$Z \vee (N \vee V)$	BHI	101000	$C = 0$ and $Z = 0$
	BLOS	101400	$C / Z = 1$			

LAMPIRAN  
 ARAHAN 2-OPERASI :

	<u>Nemonik</u>	<u>Kod Mesin</u>	<u>Kesan</u>	<u>Bit kod kondisi yang berubah</u>
DOUBLE OPERAND	MOV(B)	x1SSDD	move src to dst	N,Z set/clr by result, V clr
	ADD	06SSDD	add src to dst	all set/clr by result
	SUB	16SSDD	subtract src from dst	all set/clr by result
	CMP(B)	x2SSDD	form (src - dst)	all set/clr by result
	BIS(B)	x5SSDD	put (src OR dst) in dst	N,Z set/clr by result, V clr
	BIT(B)	x3SSDD	form (src AND dst)	N,Z set/clr by result, V clr
	BIC(B)	x4SSDD	put (- src AND dst) in dst	N,Z set/clr by result, V clr
	MUL	070RSS	multiply } result in R, and	all set/clr by result
	DIV	071RSS	divide } next reg if R odd	all set/clr by result
	XOR	074RDD	exclusive OR, result in dst	N,Z set/clr by result, V clr
PROGRAM CONTROL	JSR	004RDD	reg → stack, PC → reg, dst → PC	
	RTS	00020R	reg → PC, stack → reg	
	SPL	00023L	set CPU priority level to L	
	JMP	0001DD	dst → PC	
	SOB	077RXX	XX = offset; subtract 1 from reg. contents, if ≠ 0 branch back	
	EMT	104000 104377	PS, PC → stack, new PC, PS from 30,32	all loaded from 32
	TRAP	104400 104777	PS, PC → stack, new PC, PS from 34,36	all loaded from 36
	BPT	000003	PS, PC → stack, new PC, PS from 14,16	all loaded from 16
	RTI	000002	load PC, PS from stack	all loaded from stack
	RTT	000006	PC, PS from stack, delay T trap	all loaded from stack
MFPI	0065SS	move word from previous space to current stack	N,Z set/clr by src value, V clr	
MTPI	0066DD	move word from current stack to previous space	N,Z set/clr by result, V clr	
CPU	HALT	000000	stop CPU	
	WAIT	000001	wait for interrupt	
	RESET	000005	reset UNIBUS	

	<u>Nemonik</u>	<u>Kod</u>	<u>Syarat</u>	<u>Nemonik</u>	<u>Kod</u>	<u>Syarat</u>
CONDN. CODE OPS.	CLC	000241	clear C bit	SEC	000261	set C bit
	CLV	000242	clear V bit	SEV	000262	set V bit
	CLZ	000244	clear Z bit	SEZ	000264	set Z bit
	CLN	000250	clear N bit	SEN	000270	set N bit
	CCC	000257	clear all codes	SCC	000277	set all codes
	ASH	072RSS	SHIFT ARIT. NN PLACES (SS) NN			
			32 NN ← 31 ve ← right			

## AKSARA KAWALAN ASCII

0	NUL	Null	20	DLE	Data link escape
1	SOH	Start of heading	21	DC1	Device control 1
2	STX	Start of text	22	DC2	Device control 2
3	ETX	End of text	23	DC3	Device control 3
4	EOT	End of transmission	24	DC4	Device control 4
5	ENQ	Enquiry	25	NAK	Negative acknowledge
6	ACK	Acknowledge	26	SYN	Synchronous idle
7	BEL	Bell	27	ETB	End of transmission block
10	BS	Backspace	30	CAN	Cancel
11	HT	Horizontal tab	31	EM	End of medium
12	LF	Line feed	32	SUB	Substitute
13	VT	Vertical tab	33	ESC	Escape
14	FF	Form feed	34	FS	File separator
15	CR	Carriage return	35	GS	Group separator
16	SO	Shift out	36	RS	Record separator
17	SI	Shift in	37	US	Unit separator

KOD ASCII UNTUK AKSARA (PERLAPANAN)

40 (Space)	60	0	100	@	120	P	140	160	p
41	61	1	101	A	121	Q	141	161	q
42	62	2	102	B	122	R	142	162	r
43	63	3	103	C	123	S	143	163	s
44	64	4	104	D	124	T	144	164	t
45	65	5	105	E	125	U	145	165	u
46	66	6	106	F	126	V	146	166	v
47	67	7	107	G	127	W	147	167	w
50	70	8	110	H	130	X	150	170	x
51	71	9	111	I	131	V	151	171	y
52	72	:	112	J	132	Z	152	172	z
53	73	;	113	K	133	[	153	173	
54	74	<	114	L	134	\	154	174	
55	75	=	115	M	135	]	155	175	
56	76	>	116	N	136	^	156	176	
57	77	?	117	O	137	_	157	177 (Delete)	